Law Firm Management System

Database Design

~Author~

Kunjal Simzia

Table of Content

1. Introduction

In the contemporary legal landscape, the efficient management of legal practices has become paramount. The increasing complexity of legal work, coupled with the growing demands for accountability and transparency, necessitates the adoption of robust management systems.

The Paper-based legal practice management systems face challenges such as limited storage space, making organization difficult and retrieval time-consuming. Security risks and confidentiality concerns arise due to the vulnerability of physical documents. Manual processes lead to inefficiency, high costs for supplies, maintenance, and labor. Additionally, environmental impact and compliance difficulties emerge alongside scalability issues as the firm grows. Collaboration is limited, and version control becomes challenging, making a transition to digital or hybrid systems beneficial for improving efficiency, security, and overall management.

2. Mission Statement

Our mission is to seamlessly transition paper-based legal practice management systems into efficient, secure, and scalable digital platforms through meticulous database design.

By digitizing document storage, retrieval, and collaboration, we aim to enhance accessibility, streamline workflows, and ensure confidentiality while reducing costs and environmental impact. Our focus is on user-friendly and robust databases that enhance the performance of legal professionals to

navigate regulatory compliance with ease and adapt to evolving industry needs. Through innovative database design, we aspire to revolutionize legal practice management, optimizing efficiency and unlocking new levels of productivity for law firms of all sizes.

3. Goals and Objectives

In database design, goals and objectives are fundamental to ensuring the creation of a robust, efficient, and scalable database system. Here are some key goals and objectives defining our mission statement:

1. **Goal**: Digitize Document Storage
    - **Objective**: Implement a database structure capable of securely storing all paper documents electronically.
    - **Objective**: Develop a systematic organization scheme for easy retrieval of digitized documents.

2. **Goal**: Enhance Accessibility and Retrieval
    - **Objective**: Design a user-friendly interface for intuitive document search and retrieval.
    - **Objective**: Implement advanced search functionalities to facilitate quick access to relevant information.

3. **Goal**: Ensure Data Security and Confidentiality
    - **Objective**: Incorporate robust security measures to safeguard sensitive legal documents from unauthorized access.

- **Objective**: Establish role-based access controls to manage permissions and restrict document access as per regulatory requirements.

4. **Goal**: Facilitate seamless communication between Lawyers and Clients
    - **Objective**: Maintain transparency between client and lawyers throughout the process

5. **Goal**: Provide accurate and transparent billing and invoicing
    - **Objective**: Keeping the proper track record for all invoices and provide effortless billing

6. **Goal**: Reduce Costs
    - **Objective**: Minimize paper usage and associated costs by transitioning to digital document management.

7. **Goal**: Scalability and Adaptability
    - **Objective**: Build a scalable database infrastructure capable of accommodating the growing volume of digital documents and users.
    - **Objective**: Ensure the flexibility of the database design to adapt to evolving business needs and technological advancements.

By setting clear goals and objectives, the database design process can effectively address the specific needs and challenges of digitizing a paper-based legal practice management system, ultimately leading to improved efficiency, security, and overall effectiveness.

4. List of Subjects

Database design encompasses a wide range of subjects and concepts that are crucial for creating efficient, robust, and scalable databases. Here's a list of key subjects identified for our Law firm:

- Employee
- Customer
- Case
- Case Task
- Case Document
- Case Activity
- Case Expense
- Activity Timer
- Activity Event
- Activity Messaging
- Customer Account
- Invoice
- Invoice Line item
- Payment

5. List of Entities/Tables

In database design, entities represent the core components of the system being modeled. Each entity is typically a real-world object or concept that can have data stored about it. From the above identified subjects, list of tables has been mentioned below:

- **Employee**: The employee table is designed to store key information about employees within the law firm.
- **Customer**: The customer table is structured to capture essential details about customers for the law firm.
- **Case**: The case table is designed to manage and track individual cases within the firm, such as legal cases.
- **CaseActivity**: The case activity table is designed to log detailed activities and updates related to individual cases.
- **ActivityTask**: The activity task table is designed to record specific tasks related to the activities of a case, such as calling, emailing or events for inspection.
- **ActivityDocument**: The activity document table is designed to store documents related to specific case activities.
- **CaseExpense**: The case expense table is designed to record and track all expenses related to specific cases, such as travel.

- **Invoice**: The invoice table is structured to manage billing and payment information for transactions within a business.

- **InvoiceLineitem**: The invoice line item table records individual items listed on invoices.

- **Payment**: The payment table is essential for managing financial transactions within the firm.

- **CustomerAccount**: The funding account table serves as a repository for financial resources allocated to customers.

6. List of Attributes/Fields

In database design, attributes are properties or characteristics of entities (often represented as columns in tables). They define the data that the database will store for each entity type. Here is a comprehensive list of common attributes for every entity listed above:

1) **Employee Table**:

- **EmployeeID**: A unique identifier for each employee. It serves as the primary key for this entity.
- **FirstName**: The First name of employee.
- **LastName**: The Last name of employee.
- **JobProfile**: The job role of employee.
- **PhoneNumber**: The contact number of employee.
- **Email**: The email address of employee.
- **DateOfBirth**: The Date of birth of employee.

2) **Customer Table**:

- **CustomerID**: A unique identifier for each customer. It serves as the primary key for this entity.
- **FirstName**: The First name of customer.
- **LastName**: The Last name of customer.
- **Address**: The address of customer.
- **City**: The city customer lives in.
- **State**: The state customer lives in.
- **PostalCode**: The postal code of customer.
- **PhoneNumber**: The contact number of customer.
- **Email**: The email address of customer.

3) **Case Table**:

- **CaseID**: A unique identifier for each case. It serves as the primary key for this entity.
- **CaseNumber**: The number of case.
- **CaseName**: The name of case.
- **CaseDescription**: The full description of case.
- **OpenDate**: The date of case when opened.
- **CloseDate**: The date of case when closed.
- **Status**: The status of case.
- **PracticeArea**: The specialization in case.
- **OriginatingEmployeeID**: The employee who generated case, reference to employee table.
- **ResponsibleEmployeeID**: The employee who is handling case, reference to employee table.
- **CustomerID**: A reference to the customer associated with the case, establishing a relationship with the case.

4) **CaseActivity Table**:

- **CaseActivityID**: A unique identifier for each case activity. It serves as the primary key for this entity.
- **ActivityType**: The type of actions recorded for each case.
- **Description**: The description of activtiy.
- **BillingRate**: The cost of activity.
- **Duration**: The duration of activity.
- **IsBillable**: If it is billable or not.
- **ActivityDate**: The date when activity is done.
- **Amount**: The amount of activity.
- **CaseID**: A reference to the case associated with the CaseActivity, establishing a relationship with the CaseActivity.

5) **ActivityTask Table**:

- **TaskID**: A unique identifier for each task. It serves as the primary key for this entity.
- **Subject**: The subject of task.
- **Description**: The description of task.
- **AssignedDate**: The assigned date of task.
- **Assignee**: The person assigned to.
- **DueDate**: The last date of task.
- **Priority**: The priority of task.
- **Status**: The status of task.
- **CompletedDate**: The completion date of task.
- **CaseId**: A reference to the case associated with the ActivityTask, establishing a relationship with the ActivityTask.

6) **ActivityDocument Table**:

- **DocID**: A unique identifier for each activity document. It serves as the primary key for this entity.
- **DocName**:The name of document.

- **Description**: The description of documents.
- **UploadDate**: The date uploaded of document.
- **Author**: The author of document.
- **CaseID**: A reference to the case associated with the ActivityDocument, establishing a relationship with the ActivityDocument.

7) **CaseExpense Table**:

- **ExpenseID**: A unique identifier for each case expense. It serves as the primary key for this entity.
- **Title**: The title of expense done for the case.
- **Description**: The description of expense.
- **IsBillable**: If it is billable or not.
- **BillingRate**: The cost of expense.
- **Quantity**: The quantity of expense.
- **ExpenseDate**: The date expense is done.
- **Amount**: The amount of expense.
- **CaseID**: A reference to the case associated with the CaseExpense, establishing a relationship with the Case.

8) **Invoice Table**:

- **InvoiceID**: A unique identifier for each invoice. It serves as the primary key for this entity.
- **InvoiceNumber**: The number of invoice.
- **Description**: The description of invoice.
- **InvoiceDate**: The date invoice is generated.
- **DueDate**: The last date to pay invoice.
- **Status**: The status of invoice.
- **InvoiceAmount**: The amount of invoice.
- **CaseID**: A reference to the case associated with the CaseExpense, establishing a relationship with the Case.

- **CustomerID**: A reference to the customer associated with the Invoice, establishing a relationship with the Invoice.

9) **InvoiceLineitem Table**:

- **LineItemID**: A unique identifier for each invoice lineitem. It serves as the primary key for this entity.
- **InvoiceAmount**: The amount of invoice.
- **ActivityID**: A reference to the case activity associated with the InvoiceLineitem, establishing a relationship with the InvoiceLineitem.
- **InvoiceID**: A reference to the invoice associated with the InvoiceLineitem, establishing a relationship with the InvoiceLineitem.
- **ExpenseID**: A reference to the case expense associated with the InvoiceLineitem, establishing a relationship with the InvoiceLineitem.

10)**Payment Table**:

- **PaymentID**: A unique identifier for each payment. It serves as the primary key for this entity.
- **PaymentMethod**: The method of payment.
- **PaymentDate**: The date of payment.
- **PaymentAmount**: The amount to be paid.
- **CaseID**: A reference to the case associated with the payment, establishing a relationship with the payment.
- **InvoiceID**: A reference to the invoice associated with the payment, establishing a relationship with the payment.

11)**CustomerAccount Table**:

- **AccountID**: A unique identifier for each customer account. It serves as the primary key for this entity.

- **CustomerID**: A reference to the customer associated with the customer account, establishing a relationship with the customer account.
- **PaymentID**: A reference to the payment associated with the customer account, establishing a relationship with the customer account.
- **TotalBalance**: The total balance of account.

7. Business Rules

Business rules in database design are specific guidelines or constraints that govern various aspects of data management and usage within an organization. They are essential for ensuring data integrity, consistency, and accuracy, and they help translate real-world business requirements into database constraints and operations. Here are some key types of business rules defined according to requirements:

Employee Table:

Job profile has checklist like Sr. Lawyer, Jr. Lawyer , Head Lawyer, Administrator.

Case Table:

1. The user shall NOT be allowed to delete a Case if there is atleast one Case Activity and/or Case Expense created for the matter.
2. The user shall NOT be allowed to delete a Case if an Invoice is already created for the Case.
3. The user shall NOT be allowed to mark a Case as Closed if

- ANY of the Case activities are not complete and/or
- ALL invoices are not fully paid

4. The user shall NOT be allowed to mark a matter as Closed if
    - ANY of the activities are not complete and/or
    - ALL invoices are not fully paid
5. Close Date CANNOT be less than Open Date
6. Open date will be by default today's date

Activity Table:

1. Whenever activity is created automatically an invoice line item will be created for it.
2. Amount = (Rate * Hours/Quantity)
3. Type of the selected Case Activity:

- Task
- Event
- Call
- Email
- Documentation

Payment Table:

The user shall get the following options to select from:

1. Apply Account fund
2. Cash
3. Cheque
4. Card
5. UPI Transfer
6. Other

Invoice Table:

1. The Invoice CANNOT be modified once Approved - ONLY Status, Paid Amount and Payment Due shall be updated as needed.
2. The user shall NOT be allowed to delete a Invoice once it is "Approved"
3. All the amount of line-item table of same matter will be summed.
4. The Invoice Status have the following stages:

- Draft --> Invoice is still being prepared
- Submit for Approval -->Needs the approval so will send for approval.
- Approved --> The employee with approval rights will approve.
- Partially Paid --> this stage is autoselected if a partial payment is made for the Invoice
- Overdue --> this stage is autoselected if the Due Date has passed and FULL payment has not been made for the Invoice
- Paid --> this stage is autoselected once the Invoice is FULLY paid.

8. Relationships

In database design, relationships are crucial for defining how different entities (tables) interact with each other. They establish the logical connections between tables and are essential for organizing and structuring data in a way that reflects real-world interactions. Here's a breakdown of the main types of relationships between our entities:

1.Customer

- CustomerID: Primary Key

- **Relationships**:
  - One customer can have multiple accounts (CustomerAccount).
  - One customer can be associated with multiple cases (Case).
  - One customer can have multiple invoices (Invoice).

2.Employee

- EmployeeID: Primary Key

- **Relationships**:
  - An employee can originate multiple cases (Case.OriginatingEmployeeID).
  - An employee can be responsible for multiple cases (Case.ResponsibleEmployeeID).

3. CustomerAccount

- AccountID: Primary Key

- CustomerID: Foreign Key

- **Relationships**:
  - Multiple customer accounts can be linked to one customer.

4.Case

- CaseID: Primary Key

- CustomerID: Foreign Key

- OriginatingEmployeeID: Foreign Key

- ResponsibleEmployeeID: Foreign Key

- **Relationships**:
  - One case can be linked to one customer.
  - One case can be associated with one originating employee.
  - One case can be managed by one responsible employee.
  - One case can have multiple case activities (CaseActivity).
  - One case can have multiple activity documents (ActivityDocument).
  - One case can have multiple case expenses (CaseExpense).

- One case can have multiple tasks (ActivityTask).
- One case can generate multiple invoices (Invoice).
- One case can have multiple payments (Payment).

5.CaseActivity

- CaseActivityID: Primary Key

- CaseID: Foreign Key

- **Relationships**:
  - Multiple activities can be related to one case.
  - An activity can be linked to one invoice line item (InvoiceLineItem).

6.ActivityDocument

- DocID: Primary Key

- CaseID: Foreign Key

- **Relationships**:
  -Multiple documents can be related to one case.

7.Invoice

- InvoiceID: Primary Key

- CaseID: Foreign Key

- CustomerID: Foreign Key

- **Relationships**:
  - Multiple invoices can be linked to one case.
  - Multiple invoices can be associated with one customer.
  - One invoice can contain multiple line items (InvoiceLineItem).
  - One invoice can have multiple payments (Payment).

8.InvoiceLineItem

- LineItemID: Primary Key

- InvoiceID: Foreign Key

- ActivityID: Foreign Key

- ExpenseID: Foreign Key

- **Relationships**:
  - Multiple line items can be associated with one invoice.
  - A line item can be linked to one activity.
  - A line item can be linked to one expense.

9.CaseExpense

- ExpenseID: Primary Key

- CaseID: Foreign Key

- **Relationships**:
  - Multiple expenses can be related to one case.
  - An expense can be linked to one invoice line item (InvoiceLineItem).

10.ActivityTask

- TaskID: Primary Key

- CaseID: Foreign Key

- **Relationships**:
  - Multiple tasks can be linked to one case.

11.Payment

- PaymentID: Primary Key

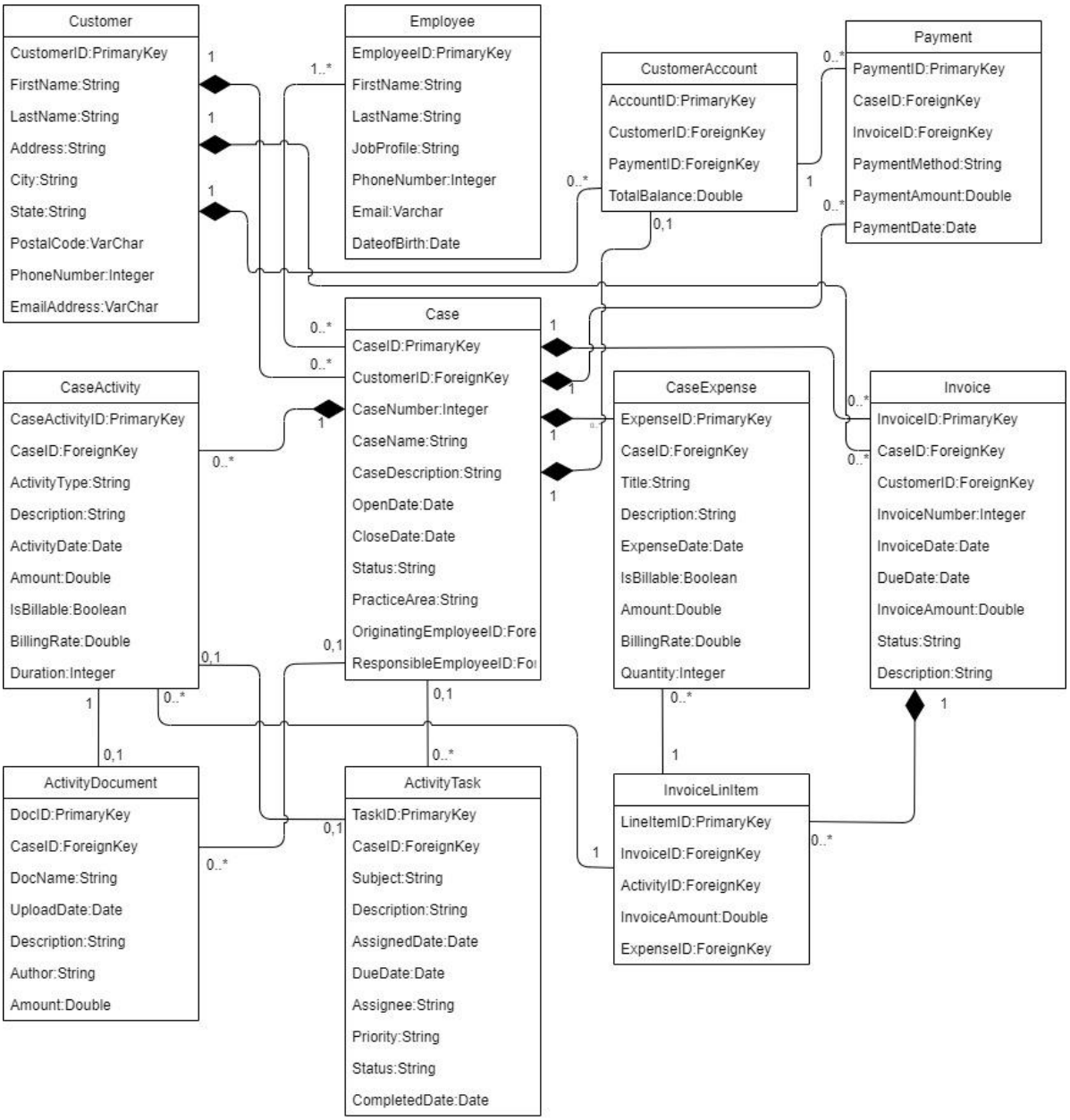
- CaseID: Foreign Key

- InvoiceID: Foreign Key

- **Relationships**:
  - Multiple payments can be linked to one case.
  - Multiple payments can be associated with one invoice.

This structured set of relationships ensures detailed and interconnected management of legal cases, capturing all necessary details, from customer data and employee involvement in financial transactions and document management.

9. Entity Relationship Diagram



10. Conclusion

The database design for a law firm management system provides a robust framework to streamline and enhance the administrative and operational functions of a law firm. By incorporating well-structured tables and clearly defined relationships, the design ensures efficient management of crucial data including employee records, customer details, case information, activities, expenses, and funding accounts.

Key features of this design include:

1. Comprehensive Data Management: The tables cover essential aspects of law firm operations, from employee and customer management to detailed tracking of cases, activities, and expenses.
2. Efficiency and Organization: The design promotes efficient data organization, facilitating easy retrieval and management of information through relationships between tables.
3. Financial Oversight: Dedicated tables for case expenses and funding accounts enable meticulous financial tracking and budget management.
4. Scalability: The modular structure allows for future expansion and integration with additional functionalities as the law firm grows and its needs evolve.
5. Data Integrity and Accuracy: The use of primary and foreign keys ensures data integrity and consistency across different tables, reducing the risk of errors and redundancies.

In summary, this database design is tailored to meet the specific requirements of law firm management, providing a solid foundation for effective case management, financial tracking, and operational efficiency. Implementing this design will help law firms optimize their workflows,

improve client service, and maintain accurate and comprehensive records, ultimately contributing to better overall management and success.

Appendix

Employee Table:

Fields	Datatype	Description
EmployeeID	INT[10]	A unique identifier for each employee
FirstName	STRING[10]	The First name of employee
LastName	STRING[10]	The Last name of employee
JobProfile	STRING[10]	The job role of employee
PhoneNumber	INT[10]	The contact number of employee
Email	VARCHAR[30]	The email address of employee
DateOfBirth	DATE	The Date of birth of employee

Customer Table:

Fields	Datatype	Description
CustomerID	INT[10]	A unique identifier for each customer
FirstName	STRING[10]	The First name of customer
LastName	STRING[10]	The Last name of customer
Address	VARCHAR[100]	The address of customer

City	STRING[10]	The city customer lives in
State	STRING[10]	The state customer lives in
PostalCode	VARCHAR[10]	The postal code of customer
PhoneNumber	INT[10]	The contact number of customer
Email	VARCHAR[30]	The email address of customer

Case Table:

Fields	Datatype	Description
CaseID	INT[10]	A unique identifier for each case
CaseNumber	INT[10]	The number of case
CaseName	STRING[10]	The name of case
CaseDescription	STRING[100]	The full description of case
OpenDate	DATE	The date of case when opened
CloseDate	DATE	The date of case when closed
Status	STRING[10]	The status of case
PracticeArea	STRING[10]	The specialization in case
OriginatingEmployeeID	INT[10]	The employee who generated case
ResponsibleEmployeeID	INT[10]	The employee who is handling case
CustomerID	INT[10]	A reference to the customer associated with the case

CaseActivity Table:

Fields	Datatype	Description
CaseActivityID	INT[10]	A unique identifier for each case activity
ActivityType	STRING[10]	The type of actions recorded for each case
Description	STRING[100]	The description of activtiy
BillingRate	CURRENCY	The cost of activity
Duration	INT[10]	The duration of activity
IsBillable	BOOLEAN	If it is billable or not
ActivityDate	DATE	The date when activity is done
Amount	DOUBLE[10,2]	The amount of activity
CaseID	INT[10]	A reference to the case associated with the CaseActivity

ActivityTask Table:

Fields	Datatype	Description
TaskID	INT[10]	A unique identifier for each task
Subject	STRING[10]	The subject of task
Description	STRING[100]	The description of task
AssignedDate	DATE	The assigned date of task
Assignee	STRING[10]	The person assigned to
DueDate	DATE	The last date of task
Priority	STRING[10]	The priority of task
Status	STRING[10]	The status of task
CompletedDate	DATE	The completion date of task

CaseId	INT[10]	A reference to the case associated with the ActivityTask

ActivityDocument Table:

Fields	Datatype	Description
DocID	INT[10]	A unique identifier for each activity document
DocName	STRING[10]	The name of document
Description	STRING[100]	The description of documents
UploadDate	DATE	The date uploaded of document
Author	STRING[10]	The author of document
CaseID	INT[10]	A reference to the case associated with the ActivityDocument

CaseExpense Table:

Fields	Datatype	Description
ExpenseID	INT[10]	A unique identifier for each case expense
Title	STRING[10]	The title of expense done for the case
Description	STRING[100]	The description of expense
IsBillable	BOOLEAN	If it is billable or not
BillingRate	CURRENCY	The cost of expense
Quantity	INT[10]	The quantity of expense
ExpenseDate	DATE	The date expense is done
Amount	DOUBLE[10,2]	The amount of expense

CaseID	INT[10]	A reference to the case associated with the CaseExpense

Invoice Table:

Fields	Datatype	Description
InvoiceID	INT[10]	A unique identifier for each invoice
InvoiceNumber	INT[10]	The number of invoice
Description	S	The description of invoice
InvoiceDate	DATE	The date invoice is generated
DueDate	DATE	The last date to pay invoice
Status	STRING[10]	The status of invoice
InvoiceAmount	DOUBLE[10,2]	The amount of invoice
CustomerID	INT[10]	A reference to the customer associated with the Invoice
CaseID	INT[10]	A reference to the case associated with the CaseExpense

InvoiceLineitem Table:

Fields	Datatype	Description
LineItemID	INT[10]	A unique identifier for each invoice lineitem
InvoiceAmount	DOUBLE[10,2]	The amount of invoice
ActivityID	INT[10]	A reference to the case activity associated with the InvoiceLineitem

InvoiceID	INT[10]	A reference to the invoice associated with the InvoiceLineitem
ExpenseID	INT[10]	A reference to the case expense associated with the InvoiceLineitem

Payment Table:

Fields	Datatype	Description
PaymentID	INT[10]	A unique identifier for each payment
PaymentMethod	STRING[10]	The method of payment
PaymentDate	DATE	The date of payment
PaymentAmount	DOUBLE[10,2]	The amount to be paid
CaseID	INT[10]	A reference to the case associated with the payment
InvoiceID	INT[10]	A reference to the invoice associated with the payment

CustomerAccount Table:

Fields	Datatype	Description
AccountID	INT[10]	A unique identifier for each customer account
CustomerID	INT[10]	A reference to the customer associated with the customer account
PaymentID	INT[10]	A reference to the payment associated with the customer account

TotalBalance	DOUBLE[10,2]	The total balance of account

Glossary

A

- Appendix: A supplementary section at the end of a document that provides additional information relevant to the main text.
- Attribute: A property or characteristic of an entity in a database, represented as a column in a table.

C

- Case: An individual legal matter handled by a law firm, which is tracked in the case management system.
- Case Activity: An action or event recorded in relation to a specific case, such as meetings, phone calls, or court appearances.
- Case Expense: Costs incurred in the handling of a case, such as travel, lodging, and filing fees.
- Column: A vertical entity in a table that contains data for a specific attribute.
- Customer: An individual or entity that engages the law firm for legal services, tracked in the customer table.

D

- Database: An organized collection of structured information, or data, typically stored electronically in a computer system.
- Department: A division within a law firm, often associated with specific legal practice areas, tracked in the employee table.

- Descriptive Attribute: An attribute that provides descriptive details about an entity, such as a name or description.

E

- Employee: An individual who works for the law firm, tracked in the employee table.
- Entity: A distinct object or thing in a database that can be identified and is of interest, such as an employee, customer, or case.

F

- Foreign Key: A column or set of columns in one table that uniquely identifies a row in another table, used to establish a link between the data in the two tables.
- Funding Account: A financial resource allocated to various projects or cases, tracked in the funding account table.

I

- Identifier Attribute: An attribute that uniquely identifies an entity within a table, often used as a primary key.

P

- Practice Area: A field or domain of legal practice to which a case belongs, such as family law or corporate law.

R

- Record: A single entry in a table, representing a unique instance of the entity described by the table.
- Relationship: An association between two or more tables in a database, established through foreign keys.

S

- Schema: The structure of a database, defined by tables, columns, and the relationships between them.
- Status: The current state of a case, such as open, closed, or pending.

T

- Table: A collection of related data held in a structured format within a database, consisting of columns and rows.
- Transaction: An operation or series of operations performed on a database, such as inserting, updating, or deleting records.

This glossary provides definitions for key terms used in the database design for a law firm management system, ensuring a clear understanding of the concepts and components involved.